

In the claims:

1. (currently amended) A computer system comprising:
 - at least two CPUs;
 - shared memory shared by said CPUs; and
 - at least one shared system resource accessible to said CPUs,

said shared memory having therein a resource locking table, comprising memory elements, each of said memory elements designated for being written to by only one of said at least two CPUs, and each CPU having a corresponding memory element for each shared system resource to which it has access,

wherein each of said at least two CPUs is communicatively interconnected with said shared memory and said shared system resource, and said resource locking table is operative by each of said CPUs, and

wherein said communicative interconnection is across a communications bus wherein a single read operation is capable of atomically reading at least a collection of said memory elements, said collection comprising at least two memory elements.
2. (deleted)
3. (deleted)
4. (deleted)
5. (previously amended) A computer system comprising:
 - at least two CPUs;
 - shared memory shared by said CPUs; and
 - at least one shared system resource accessible to said CPUs;

said shared memory having therein a resource locking table, comprising memory elements, each of said memory elements designated for being written to by only one of said at least two CPUs, and each CPU having a corresponding memory element for each shared system resource to which it has access,

wherein each of said at least two CPUs is communicatively interconnected with said shared memory and said shared system resource, and said resource locking table is operative by each of said CPUs, and

wherein any of said at least two CPUs can read a first collection of memory elements in a single transaction, said first collection of memory elements corresponding to requests of said CPUs for one of said at least one shared system resources.

6. (currently amended) A The computer system according to claim 5, wherein a CPU locks a shared system resource by executing control commands to accomplish the steps of:

a. checking a content of said first collection of memory elements of said shared memory and if ~~all are~~ each memory element of said first collection of memory elements is not clear, waiting for ~~all of them~~ each of said memory elements to clear;

b. setting the memory element corresponding to said CPU;

c. checking if more than one of said memory elements of said first collection of memory elements is set, and if true then resetting said memory element corresponding to said CPU and waiting for ~~all~~ each of said memory elements of said first collection of memory ~~units~~ elements to reset and repeating the sequence from step a.;

d. accessing said shared system resource corresponding to said first collection of memory elements; and

e. resetting said memory element, corresponding to said CPU, of said first collection of memory elements corresponding to said at least one shared system resource.

7. (currently amended) A The computer system according to claim 5, wherein a second collection of memory elements corresponds to a group of CPUs.

8. (currently amended) A The computer system according to claim 7, wherein a CPU locks a shared system resource by executing control commands to accomplish the steps of:

a. checking a content of said second collection of memory elements and, if not clear, waiting for them each memory element of said second collection of memory elements to ~~all~~ clear;

b. setting the memory element of said second collection of memory elements corresponding to a desired CPU group;

c. checking if more than one of said memory elements of said second collection of memory elements is set, and if true then ~~reset~~ resetting said memory element corresponding to said CPU group and waiting for ~~all~~ each of said memory elements of said second collection of memory elements to reset and ~~repeat~~ repeating the sequence from step a.,

d. checking a content of said first collection of memory elements corresponding to said shared system resource and said CPU group and if not clear, waiting for ~~them~~ each memory element of said first collection of memory elements to ~~all~~ clear;

e. setting the memory element of said first collection of memory elements corresponding to a CPU within said CPU group;

f. checking if more than one of said memory elements of said first collection of memory elements is set, and if

true then ~~reset~~ resetting said memory element corresponding to said CPU and waiting for ~~all~~ each memory element of said memory elements of said first collection of memory elements to reset and ~~repeat~~ repeating the sequence from step a.;

g. accessing said shared system resource corresponding to said first collection of memory elements;

h. resetting said memory element corresponding to said CPU, of said first collection of memory elements corresponding to said shared system resource; and

i. resetting said memory element, corresponding to said CPU group, of said second collection of memory elements corresponding to said shared system resource.

9. (deleted)

10. (currently amended) A method for locking a shared system resource for use by a single CPU the method comprising:

a. checking a content of a collection of memory elements of a resource locking table implemented in shared memory and if each memory element of said collection of memory elements is not all—are clear, waiting for ~~all~~ each of said memory elements ~~them~~ to clear;

b. setting the memory element corresponding to the CPU;

c. checking if more than one of said memory elements of said collection of memory elements is set, and if true then resetting said memory element corresponding to the CPU and waiting for ~~all~~ each of said memory elements of said collection of memory units to reset and ~~repeat~~ repeating the sequence from a;

d. accessing the resource corresponding to said collection of memory elements; and

e. resetting the memory element, corresponding to the CPU, of said collection of memory elements corresponding to said shared system resource.

11. (currently amended) A The method according to claim 10, wherein said shared memory is connected to a PCI bus.

12. (currently amended) A method for locking a shared system resource for use by a single CPU having access to a first and second collection of memory elements of shared memory, the method comprising:

a. checking a content of said second collection of memory elements and if not clear, waiting for ~~them~~ each memory element of said second collection of memory elements to all clear;

b. setting the memory element of said second collection of memory elements corresponding to a desired CPU group;

c. checking if more than one of said memory elements of said second collection of memory elements is set, and if true, resetting said memory element corresponding to said CPU group and waiting for ~~all~~ each of said memory elements of said second collection of memory elements to reset and ~~repeat~~ repeating the sequence from step a.;

d. checking a content of said first collection of memory elements corresponding to said resource and said CPU group and, if not clear, waiting for ~~them~~ each memory element of said first collection of memory elements to all clear;

e. setting said memory element of said first collection of memory elements corresponding to a CPU within said CPU group;

f. checking if more than one of said memory elements of said first collection of memory elements is set, and

if true then resetting said memory element corresponding to said CPU and waiting for all each of said memory elements of said first collection of memory elements to reset and ~~repeat~~ repeating the sequence from step a.;

g. accessing the shared system resource corresponding to said first collection of memory elements;

h. resetting said memory element, corresponding to said CPU, of said first collection of memory elements corresponding to said shared system resource; and

i. resetting said memory element, corresponding to said CPU group, of said second collection of memory elements corresponding to said shared system resource.

13. (currently amended) A The method according to claim 12, wherein said shared memory is connected to a PCI bus.

14. (currently amended) A computer program product for locking a shared system resource for use by a single CPU, the computer program product comprising:

computer readable medium having thereon software instructions for enabling a system, containing at least two CPUs and at least one shared system resource, to perform predetermined operations comprising:

a. checking a content of a collection of memory elements of a resource locking table implemented in shared memory and if ~~not all are~~ each memory element of said collection of memory elements is not clear, waiting for all each of ~~them~~ said memory elements to clear;

b. setting the memory element corresponding to said CPU;

c. checking if more than one of said memory elements of said collection of memory elements is set, and if true then resetting said memory element corresponding to said

CPU and waiting for ~~all~~ each of said memory elements of said collection of memory units to reset and ~~repeat~~ repeating the sequence from step a.;

d. ~~access~~ accessing said shared system resource corresponding to said collection of memory elements; and

e. resetting said memory element, corresponding to said CPU, of said collection of memory elements corresponding to said shared system resource.

15. (currently amended) A The computer software program product according to claim 14, wherein said CPUs and said shared memory are connected by a PCI bus.

16. (currently amended) A computer program product for locking a shared system resource for use by a single CPU having access to a first and second collection of memory elements of at least one shared memory, the computer program product comprising:

computer readable medium having thereon software instructions for enabling a system, containing at least two CPUs and at least one shared system resource, to perform predetermined operations comprising:

a. checking a content of said second collection of memory elements and if not clear, waiting for ~~them~~ each memory element of said second collection of memory elements to ~~all~~ clear;

b. setting the memory element of said second collection of memory elements corresponding to a desired CPU group;

c. checking if more than one of said memory elements of said second collection of memory elements is set, and if true then resetting said memory element corresponding to said CPU group and waiting for ~~all~~ each of said memory

elements of said second collection of memory elements to reset and ~~repeat~~ repeating the sequence from step a.;

d. checking a content of said first collection of memory elements residing in a resource locking table implemented in said shared memory and further corresponding to said shared system resource and CPU group and, if not clear, waiting for ~~them~~ each of said memory elements of said first collection of memory elements to all clear;

e. setting said memory element of said first collection of memory elements corresponding to a CPU within said CPU group;

f. checking if more than one of said memory elements of said first collection of memory elements is set, and if true then ~~reset~~ resetting said memory element corresponding to said CPU and waiting for ~~all~~ each of said memory elements of said first collection of memory elements to reset and ~~repeat~~ repeating the sequence from step a.;

g. accessing the shared system resource corresponding to said first collection of memory elements;

h. ~~Resetting~~ resetting the memory element, corresponding to said CPU, of said first collection of memory elements corresponding to said shared system resource; and

i. resetting said memory element, corresponding to said CPU group, of said second collection of memory elements corresponding to said shared system resource.

17. (currently amended) A The computer software program product according to claim 16, wherein said CPUs and said shared memory are connected by a PCI bus.